Principled Evaluation of Differentially Private Algorithms

Gerome Miklau (UMass Amherst)

joint work with
Ashwin Machanavajjhala (Duke)
Michael Hay (Colgate)
Dan Zhang (UMass Amherst)
Yan Chen (Duke)
Impressive progress (?)

Task: 2000 range queries; Dataset: trace;
Scale = 10000; domain size = 4096
Obstacles to adoption

• Privacy researchers have adopted an idealized and simplistic view of a data analyst’s workflow, often ignoring:
  
  • data representation, data cleaning, model selection, feature selection, algorithm tuning, iterative analysis.

• Practical performance of privacy algorithms is opaque to users and, in some cases, poorly understood by researchers.
  
  • The best algorithm for a task may depend on: setting of epsilon, “amount” of data, tunable algorithm parameters, data pre-processing (cleaning, representation)

  • Algorithm performance can be data-dependent because algorithms adapt or introduce bias.

• The research community lacks rigorous methodology for empirical evaluation.
Conflicting results

From Hardt et al. NIPS 2012

From Li et al. PVLDB 2014

MWEM [Hardt et al. 2012]
Privelet [Xiao et al. 2010]
Our inspiration

• Self-critique in machine learning:

• Value of benchmarks:
  • “When a field has good benchmarks, we settle debates and the field makes rapid progress.” David Patterson, CACM 2012.

• MLcomp:
  • Automated help for practitioners selecting algorithms for ML tasks.
Welcome to DPComp

Version 0.1

DPComp is a web-based tool designed to help both practitioners and researchers evaluate the accuracy of state-of-the-art differentially private algorithms.

A collaborative research project of Colgate University, Duke University, and the University of Massachusetts, Amherst.

• A set of evaluation principles.
• Tools to aid evaluation.
• A benchmark study for the task of answering workloads of range queries:
  • 15 published algorithms evaluated under ~8,000 distinct experimental configurations.
  • A companion website: dpcomp.org

Remainder of the talk

• 10 Principles
• Setup for benchmark study
• Overview of findings
• Open problems
• Our ongoing research efforts (motivated by dpcomp)
## Evaluation principles

<table>
<thead>
<tr>
<th>Diversity of inputs (Principles 1-4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diverse epsilon, diverse input data (scale, shape, domain size)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>End-to-end privacy (Principles 5-7)</th>
</tr>
</thead>
<tbody>
<tr>
<td>private pre- and post-processing; no free parameters; no side information.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sound evaluation of output (Principles 8-10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>measure error variability; measure bias; compare algorithms using inputs that result in reasonable privacy and accuracy.</td>
</tr>
</tbody>
</table>
Task: Answering range queries

Sensitive Dataset

<table>
<thead>
<tr>
<th>name</th>
<th>gender</th>
<th>nationality</th>
<th>grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice</td>
<td>Female</td>
<td>US</td>
<td>91</td>
</tr>
<tr>
<td>Bob</td>
<td>Male</td>
<td>Canada</td>
<td>84</td>
</tr>
<tr>
<td>Carlos</td>
<td>Male</td>
<td>Peru</td>
<td>82</td>
</tr>
<tr>
<td>Darmesh</td>
<td>Male</td>
<td>India</td>
<td>97</td>
</tr>
<tr>
<td>Eloise</td>
<td>Female</td>
<td>France</td>
<td>88</td>
</tr>
<tr>
<td>Faith</td>
<td>Female</td>
<td>US</td>
<td>78</td>
</tr>
</tbody>
</table>

Attributes (dimensions)

{gender, grade}

Workload of Range Queries

“Number of A female students”
(count where gender=female and grade >= 93)

“Number of C students”
(count where gender=* and 70 <= grade < 80)

...
Diverse datasets

**Principle:** Data-dependent algorithms should be evaluated on a *diverse* set of inputs

Frequency vector representation of input

```
x1    x2    x3    x4    x5    x6    ...  xn
  gender=female, grade=100
  gender=female, grade=99
  gender=female, grade=98
  ... 
  gender=male, grade=0
```
Properties:

- **domain size**: length of frequency vector
- **scale**: total number of records in database
- **shape**: the frequency vector normalized by scale.

**Desideratum**: datasets that are diverse with respect to all three properties.
Data generation
Systematically control for domain size and scale

Shape
Collect many real-world datasets

Domain size
Coarsen domain

Scale
Sample with replacement

Input: real dataset \( D \), domain \( \text{dom} \), target scale \( m \)

Frequency vector \( \mathbf{x}' \)

Frequency vector \( \mathbf{x} \)

Empirical dist. \( \mathbf{p} \)

Bucket

Normalize

Sample of scale \( m \)

Data generation table:

<table>
<thead>
<tr>
<th>Dataset name</th>
<th>Original Scale</th>
<th>% Zero Counts</th>
<th>Previous works</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adult</td>
<td>32,558</td>
<td>97.80%</td>
<td>[10, 15]</td>
</tr>
<tr>
<td>HepPh</td>
<td>147,414</td>
<td>21.17%</td>
<td>[15]</td>
</tr>
<tr>
<td>Income</td>
<td>20,787,122</td>
<td>44.97%</td>
<td>[15]</td>
</tr>
<tr>
<td>Medcost</td>
<td>9,415</td>
<td>74.80%</td>
<td>[11, 12, 17, 29]</td>
</tr>
<tr>
<td>Trace</td>
<td>25,714</td>
<td>96.61%</td>
<td>[15]</td>
</tr>
<tr>
<td>Patent</td>
<td>27,948,226</td>
<td>4.20%</td>
<td>[11, 12, 17, 29]</td>
</tr>
<tr>
<td>Search</td>
<td>335,889</td>
<td>51.03%</td>
<td>[11, 12, 17, 29]</td>
</tr>
<tr>
<td>Bids-F1</td>
<td>13,017,799</td>
<td>0%</td>
<td>new</td>
</tr>
<tr>
<td>Bids-FM</td>
<td>2,126,344</td>
<td>0%</td>
<td>new</td>
</tr>
<tr>
<td>Bids-ALL</td>
<td>7,655,502</td>
<td>0%</td>
<td>new</td>
</tr>
<tr>
<td>Md-Sal</td>
<td>135,727</td>
<td>83.12%</td>
<td>new</td>
</tr>
<tr>
<td>Md-Sal-FA</td>
<td>100,534</td>
<td>83.17%</td>
<td>new</td>
</tr>
<tr>
<td>LC-RFG1</td>
<td>5,737,472</td>
<td>61.57%</td>
<td>new</td>
</tr>
<tr>
<td>LC-RFG2</td>
<td>198,045</td>
<td>67.69%</td>
<td>new</td>
</tr>
<tr>
<td>LC-RFG-ALL</td>
<td>3,999,425</td>
<td>60.15%</td>
<td>new</td>
</tr>
<tr>
<td>LC-DTIR-F1</td>
<td>3,336,713</td>
<td>54.15%</td>
<td>new</td>
</tr>
<tr>
<td>LC-DTIR-2</td>
<td>5,914,136</td>
<td>47.30%</td>
<td>new</td>
</tr>
<tr>
<td>LC-DTIR-3</td>
<td>7,552,394</td>
<td>44.60%</td>
<td>new</td>
</tr>
<tr>
<td>LC-DTIR-4</td>
<td>8,452,314</td>
<td>42.50%</td>
<td>new</td>
</tr>
</tbody>
</table>

Data generation process:
1. Collect many real-world datasets.
2. Coarsen the domain.
3. Sample with replacement.

See paper for details.
Measuring error

**Definition 7 (Scaled Average Per-Query Error).** Let $W$ be a workload of $q$ queries, $x$ a data vector and $s = \|x\|_1$ its scale. Let $\hat{y} = K(x, W, \epsilon)$ denote the noisy output of algorithm $K$. Given a loss function $L$, we define scale average per-query error as $\frac{1}{s.q} L(\hat{y}, Wx)$.

### Example (scaled error):

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Scale</th>
<th>Absolute Error</th>
<th>Scaled Absolute Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dataset 1</td>
<td>1,000</td>
<td>100</td>
<td>0.100</td>
</tr>
<tr>
<td>Dataset 2</td>
<td>100,000</td>
<td>100</td>
<td>0.001</td>
</tr>
</tbody>
</table>

Scaled error is also error in units of a “population percentage”
Algorithms considered

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Data-independent</th>
<th>Data-dependent</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDENTITY [7]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PRIVELET [25]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>H [11]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>H_b [22]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GREEDY H [15]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNIFORM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MWEM [10]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MWEM*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AHP [29]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AHP*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DPCUBE [26]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DAWA [15]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QUADTree [4]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UGRID [21]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AGRID [21]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PHP [1]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EFPA [1]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SF [27]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Laplace mechanism

Extensions of Laplace mechanism

Noisy total count; assume uniformity.

Private partitioning; measurement over reduced domain.

2D-grid based techniques
Findings
Variation with shape

1D

Dom. size: 4096 Scale: 1k

Scaled error

DAWA MWEM* MWEM PHP EFPA DPCube AHP* SF Uniform

Algorithm

Error for a dataset
Dimensions: 1
Shape: Patent
Domain size: 4096
Scale: 1000

(\varepsilon=0.1 \text{ throughout})
Variation with shape

**1D**

Dom. size: 4096 Scale: 1k

Variation across shape

(for fixed dimension, domain size, scale)
Finding: Algorithm error varies significantly with dataset shape
**Finding**: Algorithms differ on the dataset shapes on which they perform well.

**Adult Dataset**

“Hard” for PHP, EFPA algorithms

“Easy” for DAWA, MWEM
Data-independent alternatives

Identity: Laplace noise added to frequency vector $x$

HB: hierarchy of noisy counts

[Qardaji et al. ICDE 2013]
Finding: Data-dependence can offer significant improvements in error (at smaller scales or lower epsilon).
Increasing scale
**Finding:** Some data-dependent algorithms fail to offer benefits at larger scales (or higher epsilons).
Review of Findings

• No best algorithm:
  • No single algorithm offers uniformly low error.

• Significant variation with shape
  • Algorithm error varies significantly with dataset shape and algorithms differ on the dataset shapes on which they perform well.

• Significant trade-offs with “signal strength”
  • Data-dependence can offer significant improvements in error, at smaller scales or lower epsilon values, but some data-dependent algorithms fail to offer benefits at larger scales or higher epsilons.

• Failure to beat baselines
  • Many algorithms are beaten by the IDENTITY baseline at large scales, in both 1D and 2D. At low scales, many algorithms result in error rates that are comparable to, or worse than, the Uniform baseline.
A few open questions

- Robust and private algorithm selection
  - See: Chaudhuri & Vinterbo, NIPS 2013, and our recent work “Pythia” SIGMOD 2017.
- Specialized data-dependent algorithms, or universal algorithms that can exploit structure in data?
- Error bounds for data-dependent algorithms
- Theory for non-worst case and for realistic parameters (concrete vs. asymptotic analysis)
- Richer, more complete benchmarks?